1. Build A Helloe World App using Spring

{ myapp.com/hellow-world (req)
"helloworld" (res) }

(i) doing it in the traditional spring way

step 1 → Create a maven Project

Step 2 → Develop API

(i) Creating a class anotate it with @Controller

(ii) Create a method and map the request using the annotation @GetMapping and @RequestBody (So that we can sent response to the web browser)

Step3 → Add Dependency

(i) adding ~~spring-core~~ spring-core jar dependency to the pom.xml file

(ii) adding spring-context jar dependency to the pom.xml file (Transitive dependency)

(iii) ~~If~~ I want to create a web application, and if it is a restful application, then I want to keep the spring MVC dependency

Step 5 : Add A web Server (If we want to test our application we want to deploy it on to a Server)

There are Servers like Jetty, Tomcat, JBoss etc

(i) Download the tomcat Server from the tomcat website

(ii) Add the Server to our web application

(iii) Deploy this application inside the tomcat Server

~~Dispa~~

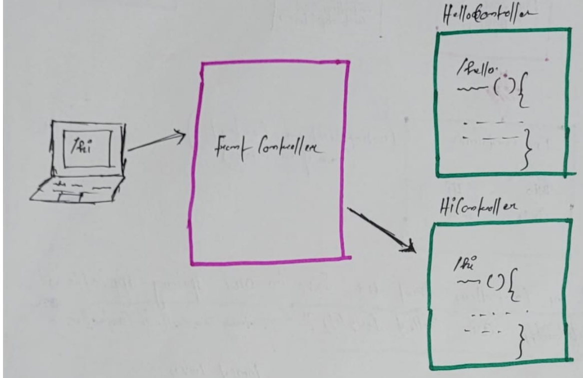
What is a front Controller ?

let say we have a Controller here Called HelloworldController which has an endpoint configured called "/hello", it does some work, ~~if com~~

we also have another Controller called HiController, it also have a url configured called "/hi"

And we are there with a laptop and hitting the url /hellow it will go to the hellow Controller. and if we hit the /hi url it will go to the hiController and this is the flow. But inbetween the client and Controller there is Front Controller

"It is the manager, this is responsible for how the request is going to be managed, where it is needed to be redirected etc" in short it act as a decision maker

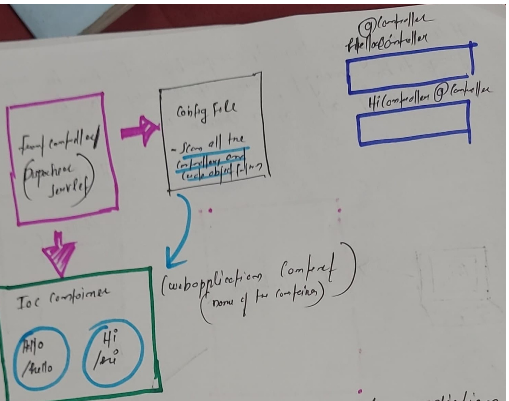


Q: How front Controller works?

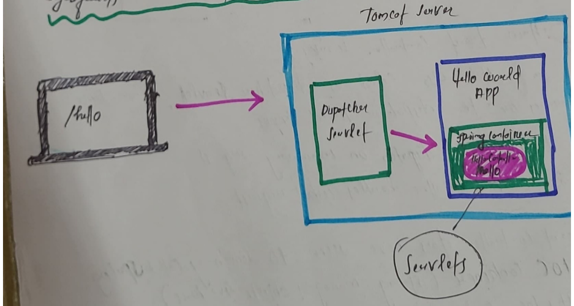
It is also called as dispatcher Servlet, It has some powers or the dispatcher servlet knows

1. where to dispatch an incomming request
2. which Controller handles what request

Front Controller does have access to something called Spring IOC container (spring managed objects live here).
This IOC Container is created by the dispatcher servlet with the help of Confi file (it contains a command like scan for all the Controllers and create the object for all of them and keep them inside IOC)



"All the Controllers that we have in our spring application bydefaults are called Servlets" (spring we call it Controller)



How many Dispatcher Servlet per application?

We can have as many front controllers in application

Step 6: write a config file

```
@Configuration
@ComponentScan ("com.ashish.helloworldapp")
Public class HelloworldConfig {

}
```

{go ahead and do a component scan inside the mentioned package and if you find any classes defected with a @Controller annotation create the bean for that and put it in the IOC container}

Step 7: Create A Dispatcher Servlet.

By extending the class (AbstractAnnotationConfigDispatcherServlet-Initializer)

(! Also add dependencies of Jakarta Servlet api)